PR #27201 完整报告

sgl-project/sglang

[AMD][WA] force to use gate_mode interleaved to fix tp2/tp4/tp8 acc issue

合并时间：2026-06-06 11:18

原文链接：http://prhub.com.cn/sgl-project/sglang/pull/27201

执行摘要

- 一句话：强制 interleave MoE 布局修复 AMD TP>1 精度崩溃
- 推荐动作：建议精读。该 PR 展示了如何快速定位硬件后端内核 bug、设计 workaround 并验证精度恢复的完整流程，对于处理类似跨平台兼容性问题有参考价值。同时注意 .to(torch.int32) 类的问题在类型敏感性高的系统中很典型。

功能与动机

gpt-oss-120b GSM8K accuracy collapses to ~0.04 (flexible-extract) on TP=2/4/8 when the AITER backend is enabled. Root cause: the FlyDSL flydsl_moe1_afp4_wfp4_bf16_t*_fp4 kernel is incorrect when used with SEPARATED layout at TP≥2. This PR works around the kernel bug by switching to the INTERLEAVE path.

实现拆解

1. 切换 weight shuffle 路径（mxfp4.py）：在 AITER 分支中，将 shuffle_weight(is_guinterleave=False, gate_up=True) 替换为 shuffle_weight_a16w4，同时将 shuffle_scale 替换为 shuffle_scale_a16w4。这样产生的 tile 布局匹配 INTERLEAVE 内核（即 bf16 MoE 内核），完全绕过有问题的 fp4x2 FlyDSL 内核。
2. 更新服务器默认值（server_args.py）：注释掉原来强制设置 SGLANG_USE_AITER_MOE_GU_ITLV=False 的语句，使得环境变量默认值生效（默认为 True，即 INTERLEAVE）。
3. 修复类型不匹配（aiter_backend.py）：在三个调用 translate_loc_from_full_to_swa 的位置（decode 图回放、prefill eager、prefill 图捕获）添加 .to(torch.int32)，因为 AITER 注意力内核（如 mha_batch_prefill_func）要求 int32 页索引，而翻译结果返回 int64。
4. 更新测试配置（test_gpt_oss_eval_mi35x.py）：将 MI35X GPT-OSS 模型测试的环境变量 SGLANG_USE_AITER_MOE_GU_ITLV 从 "0" 改为 "1"，与新的 INTERLEAVE 默认行为保持一致。

关键文件：

- python/sglang/srt/layers/quantization/mxfp4.py（模块 MoE 量化；类别 source；类型 core-logic；符号 shuffle_weight_a16w4, shuffle_scale_a16w4, process_weights_after_loading）：核心变更：切换 weight shuffle 路径从 SEPARATED

到 INTERLEAVE，直接绕过有问题的 FlyDSL 内核。

- python/sglang/srt/layers/attention/aiter_backend.py（模块 注意力后端；类别 source；类型 core-logic；符号 translate_loc_from_full_to_swa, init_forward_metadata, _apply_cuda_graph_metadata）：修复运行时类型错误：SWA 翻译结果需转为 int32 才能被 AITER 注意力内核接受。
- python/sglang/srt/server_args.py（模块 服务器配置；类别 source；类型 configuration；符号 _handle_model_specific_adjustments）：修改环境变量默认值：不再强制设置 SGLANG_USE_AITER_MOE_GU_ITLV=False，使 INTERLEAVE 成为默认。
- test/registered/amd/accuracy/mi35x/test_gpt_oss_eval_mi35x.py（模块 测试；类别 test；类型 test-coverage；符号 MI35X_GPT_OSS_MODELS）：测试配置同步：将环境变量 SGLANG_USE_AITER_MOE_GU_ITLV 从 "0" 改为 "1"。

关键符号：process_weights_after_loading, init_forward_metadata, _apply_cuda_graph_metadata

关键源码片段

python/sglang/srt/layers/quantization/mxfp4.py

核心变更：切换 weight shuffle 路径从 SEPARATED 到 INTERLEAVE，直接绕过有问题的 FlyDSL 内核。

```
# python/sglang/srt/layers/quantization/mxfp4.py
# 在 process_weights_after_loading 中，AITER 分支的核心逻辑变化：

if _use_aiter:
    # 确保 bias 为 fp32（AITER 内核要求）
    if layer.w13_weight_bias is not None:
        layer.w13_weight_bias.data = layer.w13_weight_bias.data.to(torch.float32)
    if layer.w2_weight_bias is not None:
        layer.w2_weight_bias.data = layer.w2_weight_bias.data.to(torch.float32)

    # 先做 gate/up 解交织：HF 权重存储为 [(g0, u0), (g1, u1), ...],
    # 解交织后变为 [g0, g1, ..., u0, u1, ...] 以匹配 aiter 布局期望。
    ... # 解交织代码略

    # 根据环境变量切换混洗路径：True -> INTERLEAVE（新路径），False ->
    SEPARATED（旧路径）
    if envs.SGLANG_USE_AITER_MOE_GU_ITLV.get():
        # INTERLEAVE 路径：使用 a16w4 系列函数，要求 gate/up 交织的 tile 布局
        layer.w13_weight.data = shuffle_weight_a16w4(layer.w13_weight, 16, True)
        shuffled_w13_scale = shuffle_scale_a16w4(
            layer.w13_weight_scale.view(-1, layer.w13_weight_scale.shape[-1]),
            self.num_experts,
            True,
        )
        layer.w2_weight.data = shuffle_weight_a16w4(layer.w2_weight, 16, False)
        shuffled_w2_scale = shuffle_scale_a16w4(
```

```
            layer.w2_weight_scale.view(-1, layer.w2_weight_scale.shape[-1]),
            self.num_experts,
            False,
        )
    else:
        # SEPARATED 路径（原默认，现已修复但保留）：使用标准混洗，is_guinterleave=False
        layer.w13_weight.data = shuffle_weight(
            layer.w13_weight, is_guinterleave=False, gate_up=True
        )
        shuffled_w13_scale = shuffle_scale(
            layer.w13_weight_scale.view(-1, layer.w13_weight_scale.shape[-1]),
            experts_cnt=self.num_experts,
            is_guinterleave=False,
            gate_up=True,
        )
        layer.w2_weight.data = shuffle_weight(
            layer.w2_weight, is_guinterleave=False, gate_up=False
        )
        shuffled_w2_scale = shuffle_scale(
            layer.w2_weight_scale.view(-1, layer.w2_weight_scale.shape[-1]),
            experts_cnt=self.num_experts,
            is_guinterleave=False,
            gate_up=False,
        )
```

python/sglang/srt/layers/attention/aiter_backend.py

修复运行时类型错误：SWA 翻译结果需转为 int32 才能被 AITER 注意力内核接受。

```
# python/sglang/srt/layers/attention/aiter_backend.py
# 三处调用 translate_loc_from_full_to_swa 后增加类型转换

# 1. decode 图回放路径（约第 905 行）
if self.use_sliding_window_kv_pool:
    # AITER attention kernels require int32 page indices;
    # full_to_swa_index_mapping is stored as int64.
    swa_page_table = (
        self.token_to_kv_pool.translate_loc_from_full_to_swa(
            kv_indices
        ).to(torch.int32) # 新增 .to(torch.int32)
    )

# 2. prefill eager 路径（约第 1385 行）
if self.use_sliding_window_kv_pool:
    # AITER attention kernels (e.g. mha_batch_prefill_func)
    # require int32 page indices; full_to_swa_index_mapping is
    # stored as int64.
    swa_page_table = (
        self.token_to_kv_pool.translate_loc_from_full_to_swa(
            self.indices_updater_prefill.kv_indices
```

```
        ).to(torch.int32) # 新增 .to(torch.int32)
    )

# 3. 图模式元数据应用（约第 1585 行）
if self.use_sliding_window_kv_pool:
    # AITER attention kernels require int32 page indices;
    # full_to_swa_index_mapping is stored as int64.
    swa_page_indices = (
        self.token_to_kv_pool.translate_loc_from_full_to_swa(
            page_indices
        ).to(torch.int32) # 新增 .to(torch.int32)
    )
```

评论区精华

PR 描述本身提供了详尽的分析和基准数据。合并者 HaiShaw 批准了 PR，并在评论中要求将另一个不相关的测试失败单独处理（test_mori_transfer_engine_e2e.py），表示这个 workaround 优先合入。没有其他 review 争议。

- 暂无高价值评论线程

风险与影响

- 风险：
  1. 性能回退风险：切换到 INTERLEAVE 后 *decode* 步骤默认使用 bf16 MoE 内核，但作者在 TP=1 上实测性能差异 <1%，TP≥2 的基准也已恢复正常。
  2. 类型转换覆盖不全：三处 .to(torch.int32) 已覆盖 *decode* 和 prefill 的图模式与非图模式；但如果未来新增 SWA 调用点而未注意类型要求，可能再次出现运行时错误。
  3. AITER 内核依赖：本 PR 是 workaround，底层 FlyDSL 内核 bug 本身未修复；若后续 aiter 库修复了 SEPARATED 路径，需重新评估是否切回。
  4. 环境变量兼容性：SGLANG_USE_AITER_MOE_GU_ITLV 默认值变化，可能影响其他手动设置该变量的用户（但影响是正向的，因为 INTERLEAVE 是预期正确的路径）。 - 影响：影响范围：限定在 AMD ROCm 平台、使用 AITER 后端和 MXFP4 量化的 GPT-OSS 模型。精度从 0.04 恢复到 0.90（GSM8K），性能无退化。影响程度：高，修复了完全不可用的精度崩溃，且无需用户修改配置。团队：AMD 平台开发和维护者可直接使用，后续需关注 aiter 内核修复合入后可能移除 workaround。 - 风险标记：核心推理路径变更，依赖环境变量默认值，等待 AITER 内核修复

关联脉络

- PR #27091 Unify full→SWA index translation in init_forward_metadata; drop pool caches: 该 PR 引入了 translate_loc_from_full_to_swa 返回 int64 的变化，本 PR 修复其导致的类型不匹配。